What is claimed is:

1. A method for implementing a conversation between a client and a service, comprising:

receiving a message on behalf of the service;

determining a current state of the conversation;

determining valid input document types for the current state;

verifying whether the message is of one of the valid input document types for the current state; and

dispatching the message to appropriate service entry points provided by the service, until the service produces an output document of a valid output document type.

2. The method of claim 1, wherein if messages of invalid input documents types are received, further comprising raising exceptions.

3. The method of claim 1, wherein if no valid output document is produced by the service, further comprising raising exceptions.

4. The method of claim 1, further comprising formatting and returning to the client the output document in a form appropriate to the client.

5. The method of claim 1, further comprising:

calculating a new state of the conversation from the valid output document type;

determining new input document types that are valid in the new state; and

prompting for the new input document types that are valid in the new state.

6. The method of claim 1, wherein the determining the current state step includes asking the service for conversation specifications.

7. The method of claim 1, further comprising maintaining a "state" of the conversation.

8. The method of claim 1, further comprising retrieving a "state" of the conversation from the service.

9. The method of claim 1, further comprising:

calculating a new state of the conversation from the valid output document type; and

invoking client methods that can produce new input documents that are valid in the new state.

10. The method of claim 9, further comprising sending the new input documents to the service.

11. A conversation controller that implements a conversation between a client and a service, comprising:

an incoming context handler that receives a message on behalf of the service, wherein the incoming context handler is capable of parsing the message and extracting a document type of the message;

an interaction handler coupled to the incoming context handler and capable of identifying a current state, conversation specifications and the document type of the message from the message; and

a dispatch handler coupled to the interaction handler, wherein the dispatch handler parses the conversation specification and forwards the message to service entry points of the service.

12. The conversation controller of claim 11, wherein the interaction handler validates if the document type of the message is valid for the current state.

13. The conversation controller of claim 11, wherein the interaction handler calculates a new state of the conversation and new valid document types for the new state from a response returned by the service.

14. The conversation controller of claim 13, further comprising an outgoing content handler capable of constructing an outgoing message that is valid for the new state, wherein the outgoing content handler returns the outgoing message to the client.

15. The conversation controller of claim 11, further comprising a client interaction handler that dispatches a reply from the service to the client and forwards a response from the client to the service.

16. A computer readable medium comprising instructions for implementing a conversation between a client and a service, the instructions comprising:

receiving a message on behalf of the service;

determining a current state of the conversation;

determining valid input document types for the current state;

verifying whether the message is of one of the valid input document types for the current state; and

dispatching the message to appropriate service entry points of the service, until the service produces an output document of a valid output document type.

17. The computer readable medium of claim 16, further comprising formatting and returning to the client the output document in a form appropriate to the client.

18. The computer readable medium of claim 16, further comprising:

calculating a new state of the conversation from the valid output document type;

determining new input document types that are valid in the new state; and

prompting for the new input document types that are valid in the new state.

19. The computer readable medium of claim 16, wherein if messages of invalid document types are received, further comprising raising exceptions.

20. The computer readable medium of claim 16, wherein if no valid output document is produced by the service, further comprising raising exceptions.